RECEIVED
CENTRAL FAX CENTER

FEB 2 2 2007

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Michael R. Krause et al.	Examiner:	Sean M. Reilly
Serial No.:	09/980,761	Group Art Unit:	2153
Filed:	April 15, 2002	Docket No.:	10003628-2
Title:	RELIABLE MULTI-UNICAST		

CERTIFICATE OF TRANSMISSION

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

I hereby certify that the following paper(s) are being facsimile transmitted to the U.S. Patent and Trademark Office, Fax. No. (571) 273-8300 on the date shown below:

1. Response to Non-Compliant Appeal Brief (25 pgs.).

Date: February 22, 2007

PGB: cmj

By:

Name: Patrick G. Billig (Reg. No. 38,080)

26 Pages (including cover page)

FEB 22 2007

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Appellant:	Michael R. Krause et al.	Examiner: Sean M. Reilly
Serial No.:	09/980,761	Group Art Unit: 2153
Filed:	April 15, 2002	Docket No.: 10003628-2
Title:	RELIABLE MULTI-UNICAST	

RESPONSE TO NON-COMPLIANT APPEAL BRIEF

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This Response to Non-Compliant Appeal Brief is submitted in response to the Notification of Non-Compliant Appeal Brief mailed January 22, 2007.

This Appeal Brief is submitted in support of the Notice of Appeal filed on September 5, 2006, appealing the final rejection of claims 2-47 of the above-identified application as set forth in the Final Office Action mailed June 5, 2006.

At any time during the pendency of this application, please charge any required fees or credit any overpayment to Deposit Account No. 08-2025.

Appellant respectfully requests consideration and reversal of the Examiner's rejection of pending claims 2-47.

TABLE OF CONTENTS

REAL PARTY IN INTEREST

The real party in interest is Hewlett-Packard Development Company, LP having a principal place of business at 20555 S.H. 249 Houston, TX 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

RELATED APPEALS AND INTERFERENCES

There are no other appeals or interferences known to Appellant that will have a bearing on the Board's decision in the present Appeal.

STATUS OF CLAIMS

In a Final Office Action mailed June 5, 2006, claims 2-47 were finally rejected. Claims 2-47 are pending in the application, and are the subject of the present Appeal.

STATUS OF AMENDMENTS

No amendments have been entered subsequent to the Final Office Action mailed June 5, 2006.

SUMMARY OF THE CLAIMED SUBJECT MATTER

The subject matter of the independent claims involved in the Appeal is related to reliable multicasting comprising a series of replicated unicasts. The Summary is set forth as exemplary embodiments corresponding to the language of independent claims 2 and 25. Discussions about elements of claims 2 and 25 can be found at least at the cited locations in the specification and drawings.

One aspect of the present invention, as claimed in independent claim 2, provides a distributed computer system (*See Figure 13, distributed computer system 600 and specification at page 33, line 14*) comprising a source endnode (*See Figure 13, source host processor endnode 602 and specification at page 33, lines 15-16*) participating in a multicast group. The source endnode includes a source process (*See Figure 13, source process 610 and specification at page 33, line 17*) which produces message data, a send work queue (*See*

Figure 13, QP 630 and specification page 33, lines 22-23) having work queue elements that describe the message data for multicasting, and a network interface controller (*See Figure 13, SANIC 618 and specification at page 33, line 19*) having a completion processing unit (*See Figure 13, completion processing unit 680 and specification at page 36, line 12-17*) configured to generate a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode. The distributed computer system comprises multiple destination endnodes (*See Figure 13, destination host processor endnodes 604, 606, and 608 and specification at page 33, lines 15-16*) participating in the multicast group. Each destination endnode includes a destination process (*See Figure 13, destination processes 612, 614, and 616 and specification at page 33, lines 17-19*) and a receive work queue (*See Figure 13, QPs 638, 642, and 646 and specification at page 33, lines 24-27*) having work queue elements that describe where to place incoming message data. The distributed computer system comprises communication fabric (*See Figure 13, switch 648, link 650, link 656, switch 658, link 660, link 666, and link 668 and specification at page 34, lines 10-20*) providing communication between the source endnode and the multiple destination endnodes. The distributed computer system comprises multiple end-to-end contexts (*See Figure 13, end-to-end (EE) context 631a, and EE context 631b, EE context 632a and EE context 632b, EE context 634a and EE context 634b and specification at page 33, line 28 through page 34, line 9*). Each end-to-end context has a portion storing state information at the source node and a portion storing state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes. A reliable multicast comprises a series of replicated unicasts of message data though the send work queue and each of the end-to-end contexts portions at the source endnode to the receive work queue and the corresponding end-to-end context portion at each of the destination endnodes. *See generally specification at page 33, line 12 through page 43, line 17; and Figures 13-15.*

One aspect of the present invention, as claimed in independent claim 25, provides a method of multicasting in a distributed computer system (*See Figure 13, distributed computer system 600 and specification at page 33, line 14*) including a source endnode (*See*

Figure 13, source host processor endnode 602 and specification at page 33, lines 15-16) participating in a multicast group and multiple destination endnodes (*See Figure 13, destination host processor endnodes 604, 606, and 608 and specification at page 33, lines 15-16*) participating in the multicast group. The method comprises: producing message data with a source process (*See Figure 13, source process 610 and specification at page 33, line 17*) at the source endnode; describing the message data for multicasting with work queue elements in a send work queue (*See Figure 13, QP 630 and specification page 33, lines 22-23*) at the source endnode; describing where to place incoming message data with work queue elements in a receive work queue (*See Figure 13, QPs 638, 642, and 646 and specification at page 33, lines 24-27*) at each of the multiple destination endnodes; storing in each of multiple end-to-end contexts (*See Figure 13, end-to-end (EE) context 631a, and EE context 631b, EE context 632a and EE context 632b, EE context 634a and EE context 634b and specification at page 33, line 28 through page 34, line 9*) state information at the source node and state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes; reliably multicasting data including performing a series of replicated unicasts of message data though the send work queue and each of portions of the end-to-end contexts at the source endnode to the receive work queue and corresponding end-to-end context portions at each of the destination endnodes; and generating, with a network interface controller (*See Figure 13, SANIC 618 and specification at page 33, line 19*) at the source endnode, a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode. *See generally specification at page 33, line 12 through page 43, line 17; and Figures 13-15.*

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

I. Claims 2-18, 22, 25-41, and 45 stand rejected under 35 U.S.C. §103(a) as being unpatentable over the Request for Comment 793, Transmission Control Protocol (Sept. 1981) (RFC 793) reference and the P.V. Mockapetris, Analysis of Reliable Multicast Algorithms for Local Networks, ACM (1983) reference and the Block et al. U.S. Patent No. 6,192,417.

II. Claims 19-21 and 42-44 stand rejected under 35 U.S.C. §103(a) as being unpatentable over the RFC 793 reference and the Mockapetris reference and the Block et al. U.S. Patent No. 6,192,417, and further in view of the J.M. Aldrich (USNET post, Oct. 16, 1997) reference.

III. Claim s 23-24 and 46-47 stand rejected under 35 U.S.C. §103(a) as being unpatentable over the RFC 793 reference and the Mockapetris reference and the Block et al. U.S. Patent No. 6,192,417, and further in view of the Request for Comment 2236, Internet Group Management Protocol, Version 2 (Nov. 1997) (RFC 2236) reference.

ARGUMENT

I. The Applicable Law

With regard to a 35 U.S.C. § 103 obviousness rejection: "Patent examiners carry the responsibility of making sure that the standard of patentability enunciated by the Supreme Court and by the Congress is applied in each and every case." M.P.E.P. 2141 (emphasis in the original). The Examiner bears the burden under 35 U.S.C. § 103 in establishing a *prima facie* case of obviousness. *In re Fine*, 837 F.2d 1071, 1074, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988).

Three criteria must be satisfied to establish a *prima facie* case of obviousness. First, the Examiner must show that some objective teaching in the prior art or some knowledge generally available to one of ordinary skill in the art would teach, suggest, or motivate one to modify a reference or to combine the teachings of multiple references. *In re Fine* at 1074. Second, the prior art can be modified or combined only so long as there is a reasonable expectation of success. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 USPQ 375, 379 (Fed. Cir. 1986). Third, the reference or combined references must teach or suggest all of the claim limitations. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (C.C.P.A. 1974).

The court in *Fine* stated:

> Obviousness is tested by "what the combined teaching of the references would have suggested to those of ordinary skill in the art." But it "cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination." And "teachings of references can be combined *only* if there is some suggestion or incentive to do so."

In re Fine, 5 USPQ2d at 1599 (citations omitted).

There must be some teaching somewhere that provides the suggestion or motivation to combine prior art teachings and applies that combination to solve the same or similar problem that it addresses. *In re Nilssen*, 851 F.2d 1401, 1403, 7 USPQ2d 1500, 1502 (Fed. Cir. 1988); *In re Wood*, 599 F.2d 1032, 1037, 202 USPQ 171, 174 (C.C.P.A. 1979). In particular, "The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art, and not based upon Appellant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991); M.P.E.P. § 2142 (emphasis added).

The test for obviousness under § 103 must take into consideration the invention as a whole; that is, one must consider the particular problem solved by the combination of elements that define the invention. *Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1143, 227 USPQ 543, 551 (Fed. Cir. 1985). Furthermore, claims must be interpreted in light of the specification, claim language, other claims, and prosecution history. *Panduit Corp. v. Dennison Mfg. Co.*, 810 F.2d 1561, 1568, 1 USPQ2d 1593, 1597 (Fed. Cir. 1987), *cert. denied*, 481 U.S. 1052 (1987). At the same time, a prior patent cited as a § 103 reference must be considered in its entirety, "*i.e.* as a *whole*, including portions that lead away from the invention." *Id.* That is, the Examiner must recognize and consider not only the similarities, but also the critical differences between the claimed invention and the prior art as one of the factual inquiries pertinent to any obviousness inquiry under 35 U.S.C. § 103. *In re Bond*, 910 F.2d 831, 834, 15 USPQ2d 1566, 1568 (Fed. Cir. 1990) (emphasis added). Finally, the Examiner must avoid hindsight. *Id.*

With regard for the test for obviousness under § 103, a statement that modifications of the prior art to meet the claimed invention would have been " 'well within the ordinary skill of the art at the time the claimed invention was made' " because the references relied upon teach that all aspects of the claimed invention were individually known in the art is not sufficient to establish a *prima facie* case of obviousness without some objective reason to combine the teachings of the references. *Ex parte Levengood*, 28 USPQ2d 1300 (Bd. Pat. App. & Inter. 1993); M.P.E.P. § 2143.01 (emphasis in the original).

In conclusion, an Appellant is entitled to a patent grant if any one of the elements of a prima facie case of obviousness is not established. The Federal Circuit has endorsed this view in stating: "If examination at the initial stage does not produce a prima facie case of unpatentability, then without more the Appellant is entitled to grant of the patent." In re Oetiker, 977 F.2d 1443, 1446, 24 USPQ2d 1443, 1448 (Fed. Cir. 1992).

II. Rejection of Claims 2-18, 22, 25-41, and 45 under 35 U.S.C. §103(a) as being unpatentable over the Request for Comment 793, Transmission Control Protocol (Sept. 1981) (RFC 793) reference and the P.V. Mockapetris, Analysis of Reliable Multicast Algorithms for Local Networks, ACM (1983) reference and the Block et al. U.S. Patent No. 6,192,417.

Independent claim 2 includes the limitations that the source endnode participating in the multicast group includes "a network interface controller having a completion processing unit configured to generate a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode."

Independent claim 25 includes "generating, with a network interface controller at the source endnode, a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode."

The Examiner admits that neither that RFC 793 reference nor that Mockapetris reference teach or suggest the limitations of independent claim 1 of a completion processing unit configured to generate a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode. The Examiner also admits that neither that RFC 793 reference nor that Mockapetris reference teach or suggest the limitations of independent claim 25 of generating a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode. The Examiner relies on the Block patent to teach these limitations.

The Block patent discloses a cluster topology servicer 124, a cluster communication servicer 125, and a network message servicer 126 **which are computer programs stored in main memory 120, which can be integrated with the operating system or be provided as add on applications to operating systems**. Cluster topology servicer 124 provides the functionality needed to set up and implement one or more multicast groups in a cluster. The cluster communication servicer 125 acts as both a sender of messages to other nodes

(including multicast messages to predefined groups); and a receiver of messages from other nodes. The a network message servicer 126 comprises a protocol suite for sending and receiving multicast and point to point messages as directed by cluster communication servicer 125.

Thus, the Block patent does not teach or suggest the limitations of independent claim 1 of the source endnode participating in the multicast group including "a network interface controller having a completion processing unit configured to generate a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode." The Block patent also does not teach or suggest the limitations of independent claim 25 of "generating, with a network interface controller at the source endnode, a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode."

In addition, the RFC 793 reference does not teach or suggest the limitations in independent claim 2 of multiple end-to-end contexts, each end-to-end context having a portion storing state information at the source node and a portion storing state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes, wherein a reliable multicast comprises a series of replicated unicasts of message data through the send work queue and each of the end-to-end context portions at the source endnode to the receive work queue and the corresponding end-to-end context portion of each of the destination endnodes.

The RFC 793 reference also does not teach or suggest the method of independent claim 25 of multicasting in a distributed computer system including a source endnode participating in a multicast group and multiple destination endnodes participating in the multicast group including storing in each of multiple end-to-end contexts state information at the source node and state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes, and reliably multicasting data including performing a series of replicated unicasts of message data to the send work queue and each of

portions of the end-to-end contexts at the source endnode to the receive work queue and corresponding end-to-end context portions of each of the destination endnodes.

As admitted by the Examiner, the RFC 793 reference fails to disclose a reliable multicast to a group of destination endnodes wherein the reliable multicast comprises a series of replicated unicasts to each endnode.

In addition, the RFC 793 reference teaches the transmission control protocol (TCP) which employs a reliable connection service between two processes. A similar reliable connection service to communicate between distributed processes is illustrated in Figure 3 and described from page 12, line 20-page 14, line 9 of the Present Specification. The TCP reliable connection service and the reliable connection service described and illustrated in the Present Specification both require an association of a local send buffer or queue and receive buffer or queue (i.e., queue pair (QP)) with one and only one remote QP. In a reliable connection service a non-sharable resource connection must be established between a source process and a destination process. The connection establishment and clearing of the TCP reliable connection service is described in Section 2.7, beginning at page 10 of the RFC 793 reference, which states that a connection is fully specified by the pair of sockets at the ends, and the connection can be used to carry data in both directions, that is, it is full duplex.

A reliable connection service, such as disclosed in the RFC 793 reference and disclosed in the Present Specification, requires a process to create a QP for each process which it is to communicate with over a network.

By contrast, the distributed computer system of independent claim 2 and the method of multicasting in a distributed computer system of independent claim 25 employ multiple end-to-end contexts, each having a portion storing state information at the source node and a portion storing state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes, wherein a reliable multicast comprises a series of replicated unicasts of message data through the send work queue and each of the end-to-end context portions at the source endnode to the receive work queue and the corresponding end-to-end context portions at each of the destination endnodes. Embodiments of reliable multi-unicast services are described in detail beginning at page 33, line 12 and correspondingly illustrated in Figures 13-15 of the Present Specification.

The end-to-end contexts at the source endnode and the destination endnodes as recited in independent claims 2 and 25 permit mupltiple reliable unicast services, such as reliable datagram services, to be established between the source process and the destination processes. The established reliable unicast services, such as reliable datagram services, are effectively connectionless as a result of the end-to-end contexts at the source endnode and the destination endnodes, which can greatly improve scalability.

Moreover, the Mockapetris reference does not teach or suggest the limitations of independent claim 2 of a reliable multicast comprising a series of replicated unicasts of message data through the send work queue and each of the end-to-end context portions at the source endnode to the receive work queue and the corresponding end-to-end context portion at each of the destination endnodes.

The Mockapetris reference also does not teach or suggest the limitations of independent claim 25 of reliably multicasting data including performing a series of replicated unicasts of message data through the send work queue and each of portions of the end-to-end contexts at the source endnode to the receive work queue and corresponding end-to-end context portions at each of the destination endnodes.

By contrast, the Mockapetris reference teaches simulation of a multicast in a local network which lacks any sort of one-to-many transmission capability. In the Mockapetris reference, the sender merely transmits separate messages to each destination and receives separate acknowledgements in return. Each destination requires two transmissions, so that a multicast set of N destinations require 2 N transmissions. Each transmission is created and received by a host, so a grand total of 4 N packets are processed by all hosts. As stated in the Mockapetris reference at page 153, column 2, this simulated multicast is "usually the most expensive."

Thus, the Mockapetris simulated multicast is a straightforward way for a local network which lacks any sort of one-to-many transmission capabilities to simulate a multicast, but the method is "usually the most expensive" to implement.

By contrast, the distributed computer system of independent claim 2 and the method of multicasting in a distributed computer system of independent claim 25 employ multiple end-to-end contexts, each end-to-end context having a portion storing state information at the source node and a portion storing state information at a corresponding one of the destination

endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes to provide a reliable multicast which has one-to-many transmission capability (i.e., not just a simulated expensive multicast as taught in the Mockapetris reference) which comprises a series of replicated unicasts of message data through the send work queue and each of the end-to-end contexts portions of the source endnode to the receive work queue and the corresponding end-to-end context portion at each of the destination endnodes.

Therefore, the RFC 793 reference, the Mockapetris reference, and the Block patent do not teach or suggest alone or in combination all of the limitations of the distributed computer system of independent claim 2 or all of the limitations of the method of independent claim 25.

In addition, dependent claims 3-18, and 22 further define patentably distinct independent claim 2. Dependent claims 26-41 and 45 further define patentably distinct independent claim 25. Therefore, these dependent claims are also believed to be allowable.

Therefore, Appellants respectfully request reversal of the rejection of claims 2-18, 22, 25-41, and 45 under 35 U.S.C. § 103(a) and allowance of these claims.

III. Rejection of Claims 19-21 and 42-44 under 35 U.S.C. §103(a) as being unpatentable over the RFC 793 reference and the Mockapetris reference and the Block et al. U.S. Patent No. 6,192,417, and further in view of the J.M. Aldrich (USNET post, Oct. 16, 1997) reference.

In view of the above, independent claim 2 and independent claim 25 are believed to be allowable over the cited references. Dependent claims 19-21 further define patentably distinct independent claim 2 and dependent claims 42-44 further define patentably distinct independent claim 25. Therefore, these dependent claims are also believed to be allowable.

Therefore, Appellants respectfully request reversal of the rejection of claims 19-21 and 42-44 under 35 U.S.C. § 103(a) and allowance of these claims.

IV. Rejection of Claims 23-24 and 46-47 under 35 U.S.C. §103(a) as being ~~unpatentable over the RFC 793 reference and the Mockapetris reference and the~~ Block et al. U.S. Patent No. 6,192,417, and further in view of the Request for Comment 2236, Internet Group Management Protocol, Version 2 (Nov. 1997) (RFC 2236) reference.

In view of the above, independent claims 2 and 25 are believed to be allowable over the cited references. Dependent claims 23-24 further define patentably distinct independent claim 2. Dependent claims 46-47 further define patentably distinct independent claim 25. Therefore, these dependent claims are also believed to be allowable.

Therefore, Appellants respectfully request reversal of the rejection of claims 23-24 and 46-47 under 35 U.S.C. § 103(a) and allowance of these claims.

CONCLUSION

For the above reasons, Appellants respectfully submit that the cited references neither anticipate nor render obvious claims of the pending Application. The pending claims distinguish over the cited references, and therefore, Appellants respectfully submit that the rejections must be withdrawn, and respectfully request the Examiner be reversed and claims 2-47 be allowed.

Any inquiry regarding this Appeal Brief should be directed to either Patrick G. Billig at Telephone No. (612) 573-2003, Facsimile No. (612) 573-2005 or Kevin Hart at Telephone No. (970) 898-7057, Facsimile No. (970) 898-7247. In addition, all correspondence should continue to be directed to the following address:

IP Administration
Legal Department, M/S 35
HEWLETT-PACKARD COMPANY
P.O. Box 272400
Fort Collins, Colorado 80527-2400

Respectfully submitted,

Michael R. Krause et al.,

By their attorneys,

DICKE, BILLIG & CZAJA, PLLC
Fifth Street Towers, Suite 2250
100 South Fifth Street
Minneapolis, MN 55402
Telephone: (612) 573-2003
Facsimile: (612) 573-2005

Dated: February 22, 2007
PGB:dmw

Patrick G. Billig
Reg. No. 38,080

CERTIFICATE UNDER 37 C.F.R. 1.8:

The undersigned hereby certifies that this paper or papers, as described herein, are being transmitted via facsimile to **Fax No. (571) 273-8300** on this 22nd day of **February, 2007**.

By: ______________
Name: **Patrick G. Billig**

RECEIVED
CENTRAL FAX CENTER
FEB 2 2 2007

CLAIMS APPENDIX

1. (Cancelled)

2. (Previously Presented) A distributed computer system comprising:

a source endnode participating in a multicast group and including:

a source process which produces message data;

a send work queue having work queue elements that describe the message data for multicasting; and

a network interface controller having a completion processing unit configured to generate a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode;

multiple destination endnodes participating in the multicast group, each destination endnode including:

a destination process; and

a receive work queue having work queue elements that describe where to place incoming message data;

communication fabric providing communication between the source endnode and the multiple destination endnodes; and

multiple end-to-end contexts, each end-to-end context having a portion storing state information at the source node and a portion storing state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes, wherein a reliable multicast comprises a series of replicated unicasts of message data though the send work queue and each of the end-to-end contexts portions at the source endnode to the receive work queue and the corresponding end-to-end context portion at each of the destination endnodes.

3. (Previously Presented) The distributed computer system of claim 2 wherein the network interface controller in the source endnode is configured to packetize the message data into frames.

PAGE 17/26 * RCVD AT 2/22/2007 4:48:58 PM [Eastern Standard Time] * SVR:USPTO-EFXRF-5/1 * DNIS:2738300 * CSID:612 573 2005 * DURATION (mm-ss):05-24

4. (Previously Presented) The distributed computer system of claim 3 wherein the destination endnodes each include a network interface controller which acknowledges receipt of frames multicast from the source endnode.

5. (Previously Presented) The distributed computer system of claim 4 wherein the network interface controller and the end-to-end context portion in each destination endnode ensure strong ordering of received frames multicast from the source endnode, such that the frames are received in a same defined order as transmitted from the source endnode.

6. (Previously Presented) The distributed computer system of claim 4 wherein the source endnode retransmits frames that are not successively acknowledged in the reliable multicast.

7. (Previously Presented) The distributed computer system of claim 3 wherein the network interface controller in the source endnode includes hardware which replicates frames to be provided in the series of unicasts.

8. (Previously Presented) The distributed computer system of claim 2 wherein the source endnode includes software verbs which perform the series of unicasts as a series of individual sequenced message send operations.

9. (Previously Presented) The distributed computer system of claim 2 wherein changes in composition of the endnodes participating in the multicast group are communicated to all endnodes participating in the multicast group.

10. (Previously Presented) The distributed computer system of claim 2 wherein the source endnode and each destination endnode maintains a list of destination addresses for all other endnodes participating in the multicast group.

11. (Previously Presented) The distributed computer system of claim 4 wherein the network interface controller in each destination endnode generates cumulative acknowledgments.

12. (Previously Presented) The distributed computer system of claim 4 wherein the network interface controller in each destination endnode generates acknowledgments on a per frame basis.

13. (Previously Presented) The distributed computer system of claim 4 the network interface controller of the source endnode includes the completion processing unit which gathers acknowledgements from the destination endnodes and completes frame operation by informing the source process of an operation status of multicast frames.

14. (Previously Presented) The distributed computer system of claim 13 wherein the source endnode further comprises:

a completion queue containing information related to completed work queue elements, wherein the completion processing unit communicates with the source process via the completion queue.

15. (Previously Presented) The distributed computer system of claim 13 wherein the completion processing unit informs the source process which destination processes, if any, did not receive multicast frames.

16. (Previously Presented) The distributed computer system of claim 13 wherein the completion processing unit includes an acknowledgement counter which counts acknowledgements received from the corresponding destination endnodes in the multicast group indicating that the corresponding destination endnode has received a frame multicast from the source endnode.

17. (Previously Presented) The distributed computer system of claim 16 wherein completion processing unit generates a completion event to the source process when the

acknowledgement counter indicates that a predetermined percentage of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

18. (Previously Presented) The distributed computer system of claim 16 wherein completion processing unit generates a completion event to the source process when the acknowledgement counter indicates that all of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

19. (Previously Presented) The distributed computer system of claim 13 wherein the completion processing unit includes a bit-mask array which assigns a unique bit for each destination endnode in the multicast group and clears each bit as a corresponding acknowledgment is received from the corresponding destination endnode in the multicast group indicating that the corresponding destination endnode has received a frame multicast from the source endnode.

20. (Previously Presented) The distributed computer system of claim 19 wherein the completion processing unit generates a completion event to the source process when the bit-mask array has a predetermined percentage of bits cleared in the bit-mask array indicating that that a predetermined percentage of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

21. (Previously Presented) The distributed computer system of claim 19 wherein the completion processing unit generates a completion event to the source process when the bit-mask array has all bits cleared in the bit-mask array indicating that that all of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

22 (Previously Presented) The distributed computer system of claim 13 wherein the completion processing unit includes a timing window, wherein expiring of the timing window without necessary conditions for a completion event for a corresponding multicast frame occurring indicates that any missing acknowledgments are to be tracked and resolved.

23. (Previously Presented) The distributed computer system of claim 2 wherein a given process joins the multicast group by performing a multicast join operation.

24. (Previously Presented) The distributed computer system of claim 2 wherein a given process leaves the multicast group by performing a multicast leave operation.

25. (Previously Presented) A method of multicasting in a distributed computer system including a source endnode participating in a multicast group and multiple destination endnodes participating in the multicast group, the method comprising:

producing message data with a source process at the source endnode;

describing the message data for multicasting with work queue elements in a send work queue at the source endnode;

describing where to place incoming message data with work queue elements in a receive work queue at each of the multiple destination endnodes;

storing in each of multiple end-to-end contexts state information at the source node and state information at a corresponding one of the destination endnodes to ensure the reception and sequencing of message data multicast from the source endnode to the corresponding one of the destination endnodes;

reliably multicasting data including performing a series of replicated unicasts of message data though the send work queue and each of portions of the end-to-end contexts at the source endnode to the receive work queue and corresponding end-to-end context portions at each of the destination endnodes; and

generating, with a network interface controller at the source endnode, a completion event to the source process in response to an indication that a predetermined percentage of destination endnodes in the multicast group have reliably received a selected amount of message data multicast from the source endnode.

26. (Previously Presented) The method of claim 25 further comprising:

packetizing, at the source endnode, the message data into frames.

27. (Previously Presented) The method of claim 26 further comprising:

acknowledging, at each of the destination endnodes, receipt of frames multicast from the source endnode.

28. (Previously Presented) The method of claim 27 further comprising:
ensuring strong ordering of received frames multicast from the source endnode, such that the frames are received in a same defined order as transmitted from the source endnode.

29. (Previously Presented) The method of claim 27 further comprising:
retransmitting frames that are not successively acknowledged in the reliable multicast.

30. (Previously Presented) The method of claim 26 wherein the packetizing the message data into frames includes replicating replicatings frames to be provided in the series of unicasts.

31. (Previously Presented) The method of claim 25 wherein the series of unicasts are performed as a series of individual sequenced message send operations.

32. (Previously Presented) The method of claim 25 further comprising:
communicating changes in composition of the endnodes participating in the multicast group to all endnodes participating in the multicast group.

33. (Previously Presented) The method of claim 25 further comprsing:
maintaining, at the source endnode and each destination endnode, a list of destination addresses for all other endnodes participating in the multicast group.

34. (Previously Presented) The method of claim 27 wherein the acknowledging, at each of the destination endnodes, includes generating cumulative acknowledgments.

35. (Previously Presented) The method of claim 27 wherein the acknowledging, at each of the destination endnodes, includes generating acknowledgments on a per frame basis.

36. (Previously Presented) The method of claim 28 further comprising:

gathering, at the source endnode, acknowledgements from the destination endnodes; and

completing frame operation by informing the source process of an operation status of multicast frames.

37. (Previously Presented) The method of claim 36 further comprising:

maintaining information related to completed work queue elements in a completion queue; and

communicating with the source process via the completion queue.

38. (Previously Presented) The method of claim 36 further comrprising:

informing the source process which destination processes, if any, did not receive multicast frames.

39. (Previously Presented) The method of claim 36 further comprising:

counting acknowledgements received from the corresponding destination endnodes in the multicast group indicating that the corresponding destination endnode has received a frame multicast from the source endnode.

40. (Previously Presented) The method of claim 39 further comprising:

generating a completion event to the source process when the counted acknowledgements indicate that a predetermined percentage of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

41. (Previously Presented) The method of claim 39 further comprising:

generating a completion event to the source process when the counted acknowledgements indicate that all of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

42. (Previously Presented) The method of claim 36 further comprising:

assigning a unique bit in a bit-mask array for each destination endnode in the multicast group; and

clearing each bit in the bit-mask array as a corresponding acknowledgment is received from the corresponding destination endnode in the multicast group indicating that the corresponding destination endnode has received a frame multicast from the source endnode.

43. (Previously Presented) The method of claim 42 further comprising:

generating a completion event to the source process when a predetermined percentage of bits are cleared in the bit-mask array indicating that that a predetermined percentage of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

44. (Previously Presented) The method of claim 42 further comprising:

generating a completion event to the source process when all bits are cleared in the bit-mask array indicating that that all of the destination endnodes in the multicast group have acknowledged the multicast frame has been received.

45. (Previously Presented) The method of claim 36 further comprising:

maintaining a timing window at the source endnode, wherein expiring of the timing window without necessary conditions for a completion event for a corresponding multicast frame occurring indicates that any missing acknowledgments are to be tracked and resolved.

46. (Previously Presented) The method of claim 25 further comprising:

performing a multicast join operation to join a given process to the multicast group.

47. (Previously Presented) The method of claim 25 further comprising:

performing a multicast leave operation to remove a given process from the multicast group.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

None.